面向海量天文数据的分布式MYSQL锥形检索研究*

杨超[1]，梁波[1,2]，戴伟[1,2]，卫守林[1,2]，邓辉[3]，王锋[3]
(1. 昆明理工大学信息工程与自动化学院，云南 昆明 650000;
2. 昆明理工大学云南省计算机技术应用重点实验室，云南 昆明 650000;
3. 广州大学天体物理中心， 广东 广州 510006)

摘 要：随着巡天观测计划的进行，传统数据库技术无法满足海量天文数据的存储以及检索性能的需求。本文针对海量天文数据存储以及锥形检索时的高并发、高性能问题，采用数据库中间件技术，当海量数据到达传统数据库存储的阈值时，能够通过中间件技术以分库分表的形式存储到数据库集群中，充分整合关系性数据库和分布式技术的优点。在本文中，利用MySQL数据库集成DIF插件，在分布式数据库中建立伪球面索引，能够满足海量天文数据中锥形索引需求。

关键词：海量数据；分布式中间件；锥形检索；DIF插件；MySQL
中图分类号：TP311.13;TP391.3 **文献标识码：** **文章编号：**

0 引 言

近年来，随着大型天文观测装置的性能不断提升，以及获取天文观测数据的能力得到了空前加强，国内外涌现出大批天文观测科学工程。如我国自主设计研发的郭守敬望远镜(Large Sky Area Multi-object Fiber Spectro-scopic Telescope，LAMOST)每晚的采集光谱数据达到 20GB[2]；鲁宾天文台(Vera C.Rubin Observatory)所使用的大口径全天巡视望远镜(Large Synoptic Survey Telescope，LSST)[3]每晚采集的原始观测数据多达 15TB；平方千米射电望远镜阵(Square Kilometre Array，SKA)[1]预期每年产出约为 300PB，是国内外共同协作的大科学装置，是世界上最大的射电天文台。正是在这种大数据时代的背景下，探索海量天文数据如何高效、可扩展的检索成为天文科学观测领域内的亟待解决的难题。

科学检索传统上依赖于关系型数据库，对于非海量的结构化数据，通过建立数据索引机制，能够具有较好的检索效率。但是，随着天文观测逐渐从光学观测覆盖到全波段观测，传统数据库无法适应数据雪崩式增长所带来的冲击。NoSQL 系统通常使用键值存储格式，其中同一键下的所有数据值都存储在一起并可以快速一起访问。Brahem 等提出了 AstroSpark[4]天文检索系统；邱等实现基于分布式阵列数据库的 FASTDB 系统[5]。它们具有低延迟、可扩展、性价比高等优点。但是，由于它的底层存储格式，它缺乏关系模型的高级功能，并且损害了 ACID 属性[6]。近年来，NoSQL 演变到 NewSQL，NewSQL 系统本质上是一个关系 DBMS，具有对 SQL、索引和架构的支持，它不仅可以支持事务的 ACID 属性，而且具有与 NoSQL 相同的可伸缩性。它的基本体系结构是并行的 DBMS，在查询负载方面的性能优于 MapReduce 框架[7]。分布式关系数据库结合传统关系数据库、分布式集群以及分布式事务来实现。其不仅和传统数据库具有高度

---

* **基金项目:** 感谢国家重点研发计划（2018YFA0404603），国家自然科学基金委员会-中国科学院天文联合基金资助项目(U1831204, U1931141)，国家自然科学基金资助项目(11961141001,11903009)，广州大学"创新强校工程"项目（2017KZDXM062），云南省重点研发计划(2018IA054)，云南省应用基础研究项目(2017FB001, 2018FB103) 的资助。感谢国家天文台-阿里云天文大数据联合研究中心对本项工作的支持..
收稿日期: ； 修订日期:
作者简介: 杨超，男，硕士研究生，主要研究方向为海量天文数据存储、检索处理技术.Email:chao_yang@stu.kust.edu.cn;
戴伟， 男， 副教授.研究方向:天文技术与方法.Email:daiwei@astrolab.cn;
卫守林,男， 副教授.研究方向:天文技术与方法.Email:weishoulin@astrolab.cn;
邓辉，女，教授.研究方向:天文技术与方法.Email:denghui@gzhu.edu.cn;
王锋，教授，博导，主要研究方向为分布式数据存储与计算、物联网技术、云计算等.
通讯作者: 梁波，男，高级工程师，主要研究方向为计算机数据库技术.Email:liangbo@cnlab.net

兼容性，还可以更好的支持 SQL 和事务处理[8]。

但是，即使在性能强大的分布式关系数据库中，由于各大天文巡天项目发布的数据星表的数据量越来越大，直接检索的方式无法满足当前天文数据的检索需求，所以大规模星表数据集的检索访问需要有效的球面索引，目前应用广泛的伪球面索引方法有 HTM (Hierarchical Triangular Mesh) ，HEALPix (Hierarchical Equal Area isoLatitude Pixelisation)及 Q3C (Quad Tree Cube)[9]等。在建立适合的球面索引之后，就可以通过锥形检索来实现大规模星表数据集的检索。锥形检索作为天文领域的一种特殊检索方式[14]，其定义天区中的赤经(Right Ascension，R.A.)和赤纬(Declination，Dec)和角距离(SR)的位置信息，通常以(T_{ra}、T_{dec})为圆心，SR 为半径的锥形区域，锥形检索(Cone Search)就是对这个锥形相关天体的信息查询，即查询出目标星体(S_{ra}、S_{dec})。本文整合 DIF 索引工具到分布式 MySQL 数据集群中，使分布式关系型数据库能够建立伪球面索引从而实现高效锥形检索的需求。

本文第一节将研究数据库中间件技术的基本框架、代理模式及其高可用架构，利用分库分表和读写分离对 MySQL 数据库水平扩展，并且能够支持事务、ACID 等。第二节将研究介绍 DIF 索引工具，使用 DIF 工具来建立适合分库分表规则的伪球面索引，并且通过其内置函数实现锥形检索。第三节将研究设计试验的框架结构，以及介绍该框架下的锥形检索服务。第四节将进行 DIF 工具函数方法和大圆公式的锥形检索对比、数据库节点的对比、HEALPix 索引等级的对比、以及单机和分布式的检索性能对比分析。

1 数据库中间件

数据库中间件可以封装底层数据，实现单数据库一样的数据操作方式，有两种典型的数据库中间件的模式，分别是服务端代理(数据库代理)和客户端代理(数据源代理)。数据库代理方式是通过代理服务器，管理多个数据库实例，客户端通过数据源与代理服务器建立连接，客户端所有的SQL操作都由代理分发给底层数据库，得到的结果也同样经过代理整合返回给客户端。而数据源代理方式，是内部管理多个普通数据源，客户端的SQL操作通过数据源代理进行如SQL解析、SQL改写等，然后分发给普通数据源去执行，得到结果也是由代理合并返回给客户端[10]。从SQL处理过程来分析，数据库中间件可以解析客户端发送的事务请求，在进行比如SQL解析、优化和路由分析等，拆分为数据库可以执行的线程任务，按照预先设置好的分库分表规则分发到多个数据库服务端，可以有效地缓解单机数据库的负载压力，达到削峰的作用。为了使数据库中间件能够支持高可用，可添加配置中心和监控服务形成简单的高可用架构。通过监控服务监测到集群状态，如有变更，推送变更信息到配置中心，数据库代理Proxy拉取配置中心的配置变更信息，从而更新数据库配置。如图1所示。

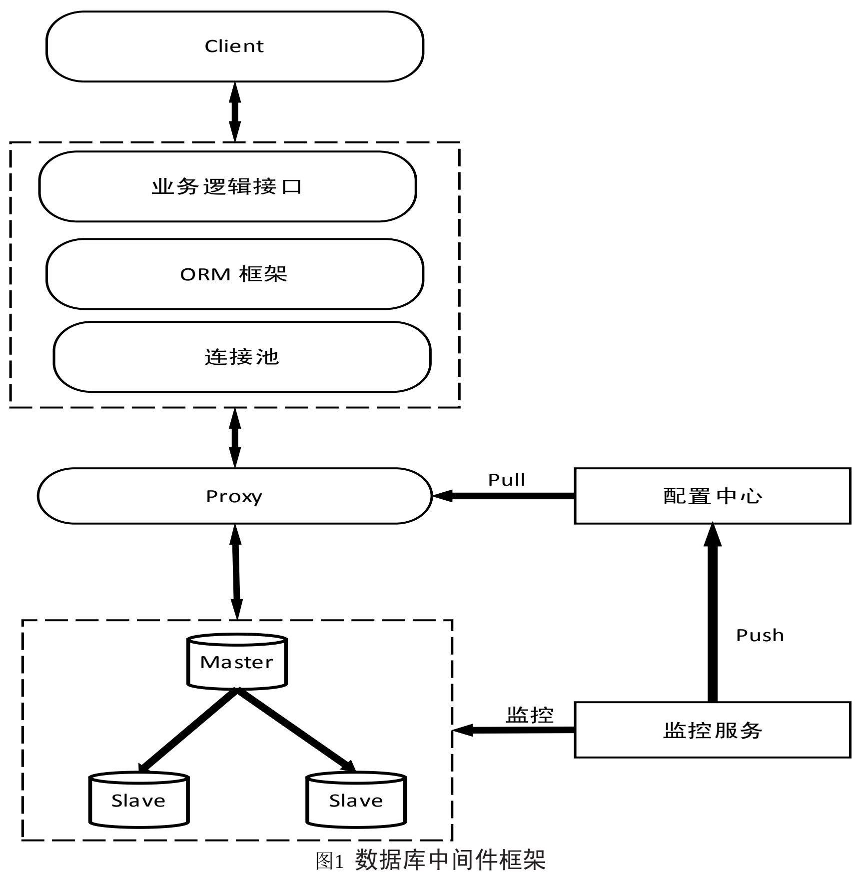


图1 数据库中间件框架

Fig.1 Database middleware architecture

为了降低锥形检索服务的负载和开销，可采用主从读写分离分布、分库分表的方式。还可以实现数据库的横向扩展，提高锥形检索效率和并发量。

1.1 分库分表

分库分表是数据库中间件的核心功能。常用切分方法是垂直切分以及水平切分，在分布式领域常用水平分库分表的方式来解决单一数据库的瓶颈，缓解单机数据库的访问压力。使用关系型数据库MySQL来水平划分，根据水平分表规则将全局关系的N表，划分为若干个不相交的子集（N_1，N_2，$N_3...N_n$）以满足完整性、重构性以及不相交性，分别对应公式 (1)，(2)，(3)[11]：

$$\text{If } a \in N \text{ is satisfied, then it can be proven that} : \text{a} \in N_i, i = 1,2...n \quad (1)$$

$$N1 \cup N2 \cup ... \cup N_j, j = 1,2...n \quad (2)$$

$$N_i, N_j \in N, \text{then } N_i \cap N_j \neq o \quad (3)$$

1.2 读写分离

由于数据库读取操作的比例远远大于写入操作，常使用一主多从的分布方式，主节点负责写操作，而从节点负责读操作。通过读写分离，将负载均衡到了多个节点上。使用读写分离的机制，需要维护好数据一致性问题，常用数据同步方式除了主从复制，还有Paxos、Raft，Term、ZAB等协议算法[12]。MySQL中常使用MySQL Proxy作为读写分离的中间层，通过内嵌式Lua解析器来定义查询处理等[13]。

2 DIF插件

动态索引工具(Dynamic Index Facility，DIF)[16]是开源的 MySQL/MariaDB 数据库插件，由 C++库、Perl 脚本、和 SQL 存储过程编译而成，其所采用的方法是“离散化”2 维空间，并使用像素化方法将其映射到 1 维空间。然后为每个像素标记一个唯一的索引 ID，如果只是按照穷举的方式进行检索，其时间复杂度为O（n）。在进行锥形检索时，需要建立伪球面索引来降低检索的时间复杂度，而使用的大多数索引都是基于树结构。B-Tree[17]索引方法可以将时间复杂度减少到$\log_2 N$，但是由于 B-Tree 的深度随着数据量的增加而增加，因此难以将其应用于海量数据的索引。基于 B^+树的索引方法是数据库系统中最常用的动态索引结构，这也是 DIF 通过 MySQL 建立伪球面索引所使用的索引方法。

DIF 索引工具使用的是目前应用最广泛的几种伪球面索引方法中的分层三角网格(Hierarchical Triangular Mesh，HTM)和 HEALPix(Hierarchical Equal Area isoLatitude Pixeli- sation，HEALPix)。HTM 作为天区划分的经典索引方式，最早应用于斯隆数字巡天数据 (Sloan Digital Sky Survey，SDSS)[9]。HEALPix 用等面积的四边形区块划分替换了 HTM 的三角形划分方式，但是它们同样具有四叉树式的层次递归规律[15]。目前，HEALPix 相关的伪球面索引建立和高效锥形检索等在海量天文数据的得到广泛应用[9]。

本文使用 DIF 索引工具预先建立 HEALPix Nest 索引，然后通过 DIF_Circle(RA，DEC，SR)函数实现锥形检索，其中 RA 为赤经、DEC 为赤纬和 SR 为角距离。DIF 是使用 healpix _base 工具包计算出 HEALPix Nest 中与圆锥相交的像素，然后，通过 MySQL 数据库查找对应像素，返回结果集。

3 设计与实现

为了实现海量天文大数据的高效锥形检索需求，本文设计了基于数据库中间件、MySQL数据库和DIF索引工具的方案。图2是设计的主体框架。

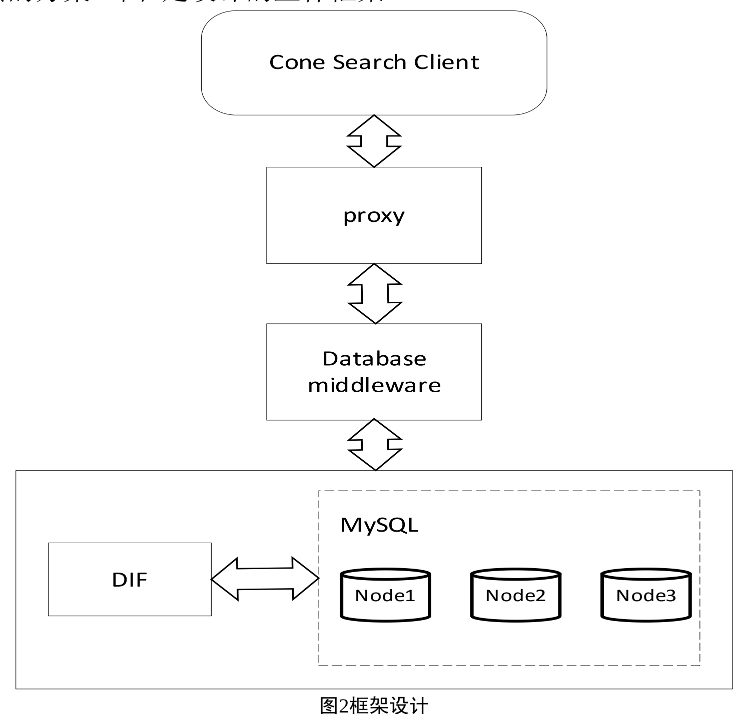


图2框架设计

Fig.2 Framework design

在图2框架中，锥形检索服务，不是直接访问底层数据库，而是先通过代理服务器按照预先设定好的策略，将锥形检索命令定向转发到数据库中间件服务，经过中间件的SQL解析、SQL路由、SQL改写、SQL执行以及结果集的合并。其中底层通过DBI/DBD-MySQL模块和MySQL数据库进行通信，可以通过MySQL数据库函数来进行调用。

4 试验及讨论

为了验证本文分布式锥形检索框架的优势，本文进行了单机关系型数据库MySQL与本文分布式架构的锥形检索对比试验。

4.1 测试环境

分布式测试平台是由中科曙光(Sugon)服务器组成集群，服务器处理器 Intel(R) Xeon(R)CPU E7- 4807，内存是 16G，存储为 2*256GB 的 SSD，服务器间通过千兆网络连接，操作系统是 Ubuntu 18.04.4 LTS。一台服务器搭建中间件系统，另外三台上搭建 MySQL 5.7.31 和 DIF 0.5.5。单机测试环境与分布式测试平台服务器相同配置。

4.2 测试数据

本测试使用 Gaia 数据集第二版的源数据表，该表包含基本的源参数。如表 1 所示，过滤原数据集一些和锥形检索无关的数据列，从中提取 source_id、ra、dec 三列数据作为测试数据，然后可以通过 DIF 工具建立 HEALPix Nest 索引。

表 1 数据集描述

Tab.1 Description of dataset

数据集	数据量	数据说明(数据参数及数据类型)		
Gaia DR2	1 ,692 ,919 ,135	source_id：唯一标识符（long）	ra: 赤经(double, 度[deg])	dec: 赤纬(double, 度[deg])

DIF 和大圆公式的锥形检索对比如表 2 所示，试验在分布式测试环境下进行，其中检索中心为(0，50)，检索半径为 1 度。第一种，DIF 工具的 SQL 模板表示以赤经、赤纬（ra，dec）为中心，SR 为角距离来进行锥形检索，其中（RA，DEC）为数据集中的赤经、赤纬，单位为度(degree，deg)，SR 的单位为角分(arcminute)。第二种，通过建立 DEC 索引，然后进行(DEC-1，DEC+1)的范围过滤，最后利用大圆公式来计算球面角距离小于 SR 的集合，从而实现锥形检索的目的，其中角距离 SR 为度(degree，deg)。大圆公式是两点赤道坐标为 p1(RA，DEC)，p2(ra，dec)，求它们的球面角距离 d，公式如（4）式[18]。

$$d = arccos[sin(DEC)sin(dec) + cos(DEC)cos(dec)cos(RA - ra)] \tag{4}$$

表 2 DIF 与大圆公式对比

Tab.2 Comparison between DIF and Great-Circle formula

查询方式	SQL	Order	Count	Intersect	Times(s)
DIF	SELECT COUNT(*) FROM GAIA_healp_nest_Order WHERE dif_Circle(ra, dec, SR);	14			4.950
大圆公式(优化)	SELECT COUNT(*) FROM GAIA WHERE DEC between DEC-SR and DEC+SR and DEGREES (ACOS(SIN(RADIANS(DEC))*SIN(RADIANS(dec)) + COS (RADIANS(DEC)) * COS (RADIANS(dec))*COS(RADIANS(RA-ra))))<SR;	无	81510	81510	890.910

由表 2 所示，两种方式的结果集数量(Count)与两结果集的交集(Intersect)数量一致，可以确定 DIF 实现方式和优化的大圆公式的结果是一致的，除此之外，DIF 查询方式的效率更高。

4.3 测试结果及分析

为了试验测试数据库节点数量以及HEALPix Nest的索引级别order对检索效率的影响，从而确定数据库节点数量和索引级别order的选用，进行了以下对比实验。由图3所示，数据库节点分别为1，3，15，30，60，100节点下锥形检索的对比，试验的检索中心是(40，0)(degree)，横轴为检索半径SR(arcmin)从30到240，竖轴为检索时间(ms)。为了消除缓存机制对实验的干扰，试验过程中通过set global query_cache_size=0或者set global query_cache_type=0来临时关闭缓存。对比发现，当节点由1增加到30节点的过程中，检索时间呈现递减趋势，然而，当节点继续增加到60甚至100的时候，检索时间呈现递增趋势。所以，本文选择30节点为试验测试节点数量。

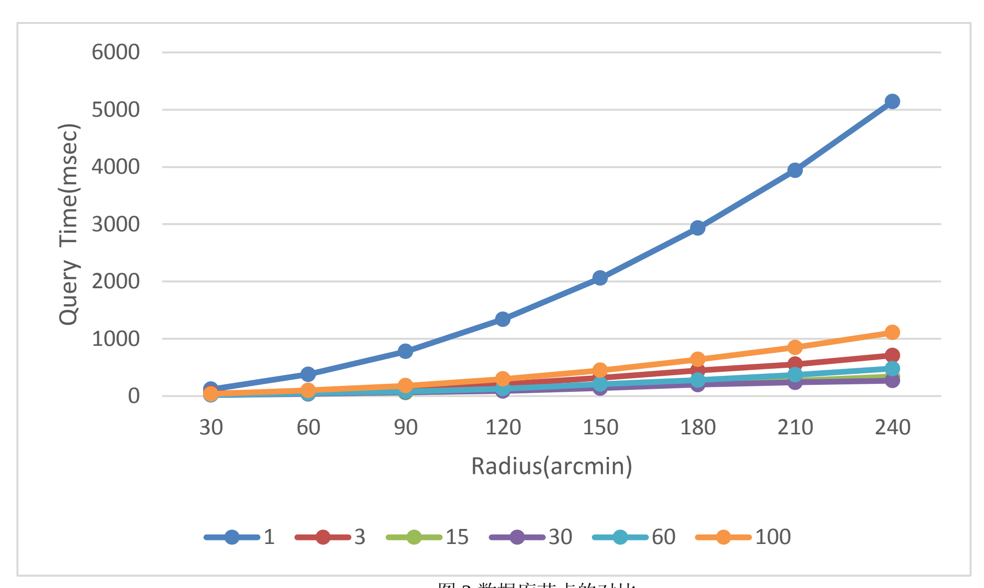


图 3 数据库节点的对比
Fig.3 Comparison for database nodes

为了对比 HEALPix Nest 各个索引等级 order 对锥形检索效率的影响，选择在 30 节点下进行(40，0)为中心，60(arcmin)为检索半径的锥形检索试验，结果如图 4 所示：

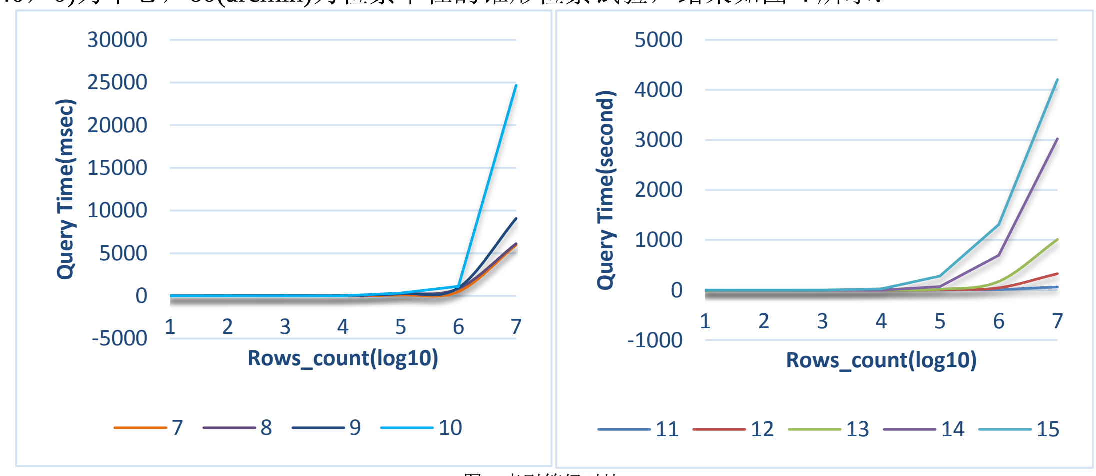


图 4 索引等级对比
Fig.4 Comparison for index level

Fan D 等[19]文献进行了索引等级与检索效率之间的对比实验，得出了索引等级 order=12 为该文献最佳方案。并指出对于不同密度的数据集，可能有不同的选择。Berriman 等[20]文献对比了该文献对比了 HTM、HEALPix 索引在 Solaris、Windows 和 Windows Server 数据库服务器的性能，提出了提高索引等级，可提高性能，但是索引等级并不是越大越好，还得考虑索引粒度过细造成的性能退化以及硬件 I/O 吞吐量等因数。本文结合文献以及图 4 的结果，选择索引等级 order 为 12 为本数据样本的最佳方案，不至于粒度过大，影响检索结果的正确率，也不至于索引等级过高造成检索效率的下降。

由上图 3 的对比，选择 30 个节点和单机通过 Jmeter 工具来进行压力测试，测试采用 500 个并发，对系统进行以(40，0)(degree)为中心，查询半径为 60(arcmin)，HEALPix 的 order 为 12，进行锥形检索，结果如下表 3、图 5 所示。

表 3 检索性能指标对比

Tab.3Comparison of retrieval performance indicators

Request	Executions		Response Time（ms）					Throughput
Label	Samples	Error%	Average	Median	90th pct	95th pct	99th pct	Transactions/s
单机	10000	0. 00%	1365. 48	1207. 00	1875. 80	2097. 40	2921. 88	57. 87
分布式	10000	0. 00%	87. 64	86. 45	134. 45	164. 32	267. 81	146. 3

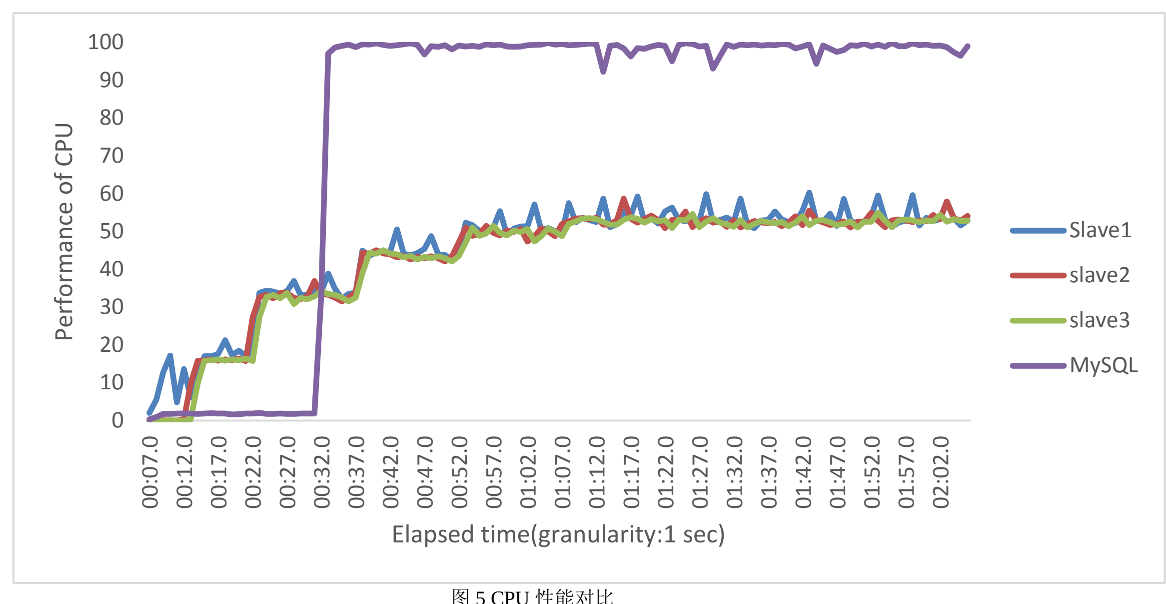


图 5 CPU 性能对比
Fig.5 Comparison for CPU performance

对比表 3，锥形检索属于数据密集型计算，十分消耗 CPU 资源，在数据库中间件的分布式方案不仅可以有效加速锥形检索效率，而且还具有更大的吞吐量。在图 5 中，MySQL 单机进行锥形检索查询的 CPU 占用率达到了 90%以上，本文中间件框架下的分布式检索的 CPU 占用率在 50%左右。本文提出的面向海量天文数据的数据库中间的分布式解决方案，能够有效提高处理锥形检索的检索效率，为未来巡天项目的锥形检索提出一些参考。

5　结束语

日益增长的海量天文数据始终面临着数据存储和高效检索的问题，本文提出了一种通过数据库中间件和DIF工具，来扩展MySQL数据库系统的方案。主要通过数据库中间件实现MySQL数据库的分库分表，来缓解面对海量天文数据存储、检索的压力。为了进一步地加快天文数据的检索，引入了DIF工具整合MySQL数据库，使该方案能够满足对海量天文数据进行高效地锥形检索的需求。通过测试，在本文中的分布式的检索方案，选择索引等级order为12，能够对海量天文数据进行高效的锥形检索，不仅可以缓解锥形检索所带来的负载压力，还能有效保证数据的安全性、数据库的高可用性。未来进一步的研究工作是对比不同的索引算法(HTM、HEALPix、Q3C等)是否同样适用于本文的架构方案，以及列式数据库或者向量引擎对锥形检索的影响。另外DIF适用于mariaDB和MySQL，需对比二者的检索性能。最后，DIF工具的算法是否可以利用GPU加速、以及如何改进锥形检索的算法，需要进一步的试验和探索。

参考文献:

[1] caife A. M. M. Big telescope, big data: towards exascale with the Square Kilometre Array. Philosophical Transactions of the Royal

Society A:Mathematical, Physical and Engineering Sciences.2020,378:2166.
[2] 汪洋,李鹏,季一木,樊卫北,张玉杰,王汝传,陈国良.高性能计算与天文大数据研究综述[J].计算机科学,2020,47(01):1-6.
[3] Abell, Allison P A, Anderson J, et al.LSST Science Book,Version 2.0[C].LSST Corporation.2009.
[4] Brahem M, Lopes S, Yeh L, et al. AstroSpark: towards a distributed data server for big data in astronomy[C]// Acm Sigspatial Phd Symposium. ACM, 2016.
[5] 邱能俊,陈梅,李晖,李宏源,黄梦琳,朱明.阵列数据库系统FASTDB的研究与实现[J].计算机工程与设计,2016,37(04):1107-1112.
[6] Grolinger K., Higashino W.A., Tiwari A., Capretz M.A.Data management in cloud environments: NoSQL and NewSQL data stores J. Cloud Comput.,2013,2 (1) :1-24.
[7] Stonebraker M , Abadi D J , Dewitt D J , et al. MapReduce and parallel DBMSs: friends or foes?[J]. Communications of the Acm, 2010, 53(1):64-71.
[8] Wang S , Zhong Y , Wang E . An integrated GIS platform architecture for spatiotemporal big data[J]. Future generation computer systems, 2019, 94(MAY):160-172.
[9] 张海龙,冶鑫晨,李慧娟,王杰,王万琼,托乎提努尔,聂俊,崔辰州,刘梁,谌俊毅,陈肖,薛岩松,何勃亮,李长华,赵青,肖健,樊东卫,曹子皇,李珊珊,米琳莹,杨哲睿.天文数据检索与发布综述[J].天文研究与技术,2017,14(02):212-228.
[10] 王政,王锋,田园,李建,赵永恒.LAMOST异构环境信息检索系统的设计与实现[J/OL].天文研究与技术:1-12[2020-10-09].
[11] 漆绍洋.基于mysql的分布式访问中间件中sql处理模块的设计与实现[D].南京大学,2016.
[12] 吴昊,陈康,武永卫,郑纬民.基于RDMA和NVM的大数据系统一致性协议研究[J].大数据,2019,5(04):89-99.
[13] 项凯.面向海量高并发数据库中间件的研究与应用[D].上海交通大学,2015.
[14] Plante, Raymond, Williams, Roy, Hanisch, Robert, etc. Simple Cone Search Version 1.03[J]. 2008.
[15] Gorski K M, Hivon E, Banday A J, et al. HEALPix: A Framework for High-Resolution Discretization and Fast Analysis of Data Distributed on the Sphere[J]. Astrophysical Journal, 2004, 622(2):759-771.
[16] Luciano N, Giorgio C. Multiple Depth DB Tables Indexing on the Sphere[J]. Advance in Astronomy,2010,2010:11.
[17] Yu L, Fu G, Jin Y, et al. MPDBS: A multi-level parallel database system based on B-Tree. Ieee/acis International Conference on Software Engineering, Artificial Intelligence, NETWORKING and Parallel/distributed Computing. IEEE 2015;1-7.
[18] 樊东卫,何勃亮,李长华,韩军,许允飞,崔辰州.球面距离计算方法及精度比较[J].天文研究与技术,2019,16(01):69-76.
[19] Fan D, Xu Y, Han J, et al. A Simple Survey of Cross-Matching Methods[J]. ASPC, 2019, 523:551.
[20] Berriman G B, Good J C, Shiao B, et al. A Study of the Efficiency of Spatial Indexing Methods Applied to Large Astronomical Databases[J]. arXiv preprint arXiv:1806.08866, 2018.

Research on Cone Search of Distributed MYSQL for Massive Astronomical Data

Yang Chao[1], Liang Bo[1,2], Dai Wei[1,2], Wei Shou Lin[1,2], Deng Hui[3], Wang Feng[3]

(1. School of Information Engineering and Automation，Kunming University of Science and Technology，Kunming，650011，China ；
2. Computer Technology Application Key Laboratory of Yunnan Province，Kunming University of Science and Technology，Chenggong 650500，China；
3. Center for Astrophysics，Guangzhou University，Guangzhou，510006，China)

Abstract: With the progress of sky survey projects, traditional database technology cannot satisfy the requirements of massive astronomical data storage and retrieval performance. In this paper, aiming at the high concurrency and performance of massive astronomical data storage and cone search, database middleware technology is used. When massive data reaches the limit of traditional database storage, it can be stored in the form of sub-database and sub-table through middleware technology. In the database cluster, the characteristics of the relational databases and distributed technology are fully integrated. In this article, the application of MySQL database integrated DIF plug-in technology to establish a pseudo-spherical index in a distributed database can meet the needs of spatial indexing in massive astronomical data.

Key words: Mass Data；Distributed Middleware；Cone Search；DIF；MySQL

* **基金项目：**中国SKA 数据中心原型建设(2018YFA0404603)；云南省重点研发计划(2018IA054); 云南省应用基础研究项目(2018FB103)资助.
收稿日期：；修订日期：
作者简介：杨超，男，硕士研究生，主要研究方向为海量天文数据存储、检索处理技术.Email:chao_yang@stu.kust.edu.cn;
戴伟，男，副教授.研究方向:天文技术与方法.Email:daiwei@astrolab.cn;
邓辉，女，教授.研究方向:天文技术与方法.Email:denghui@gzhu.edu.cn;
王锋，教授，博导，主要研究方向为分布式数据存储与计算、物联网技术、云计算等.
通讯作者：梁波，男，高级工程师，硕士，主要研究方向为计算机数据库技术.Email:liangbo@cnlab